

Authenticating users via PKI

We have already talked about the SSH Server Key, which is used to verify the server's identity and to negotiate the security (hmac/encryption) parameters. In this article, instead, we want to explain how to use PKI to authenticate users in Syncplify Server!

First of all, it is important to understand that – unlike Server Keys (or Host Keys) – these user-specific key pairs are not used for encryption, but **only and exclusively to authenticate users**, thus verifying their identity and decide whether to let them log into the server or not.

Authenticating users via PKI certainly grants a much higher degree of security than simply using a password, and is, therefore, a highly recommended authentication method.

Method #1: user-generated key pair (recommended)

To ensure the highest degree of protection for the user, and prevent the potential loss of the private key, it is highly recommended that the user generates the RSA key pair (both private and public key) and then sends only the public key to the server operator.

This can be done, for example, with the following shell command on a **Linux** client:

```
ssh-keygen -t rsa -b 4096
```

Or, in **Windows**, it's easily doable using the excellent PuTTYgen tool, as shown in the images here below:



PuTTY Key Generator

?

×

File Key Conversions Help

Key

No key.

click this

Actions

Generate a public/private key pair

Generate

Load an existing private key file

Load

Save the generated key

Save public key

Save private key

Parameters

Type of key to generate:

☐ SSH-1 (RSA)

☒ SSH-2 RSA

☐ SSH-2 DSA

Number of bits in a generated key:

2048

The screenshot shows the PuTTY Key Generator application window. The title bar reads "PuTTY Key Generator" with a question mark and a close button. The menu bar includes "File", "Key", "Conversions", and "Help".

The "Key" section contains a text area for the public key, which is highlighted in blue. The text is: `ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAQEAxJ253L+2t9D+/Dx0waF9GEgfZ4veZ4Mczeltab3GdivnvefwR8cv/x1ZGM2S91w+GPvTHHVezHXLall88ZlxmuGxi8JaSnX1dqkPmQuk0+27+MXy0V8laBynpsb6dHqIQAVOZ3iapTk+v35+ui2LFgvHjsA6j68p0wozFtZgWvK76Jf/crrfXd8/xQsElv8ZzGOFRkiUJd/u2bh`. Below this are fields for "Key fingerprint" (displaying "ssh-rsa 2048 db:6e:4f:c0:8b:44:8d:e0:0f:8f:7a:9d:93:d8:8e:21"), "Key comment" (displaying "rsa-key-20200710"), "Key passphrase", and "Confirm passphrase".

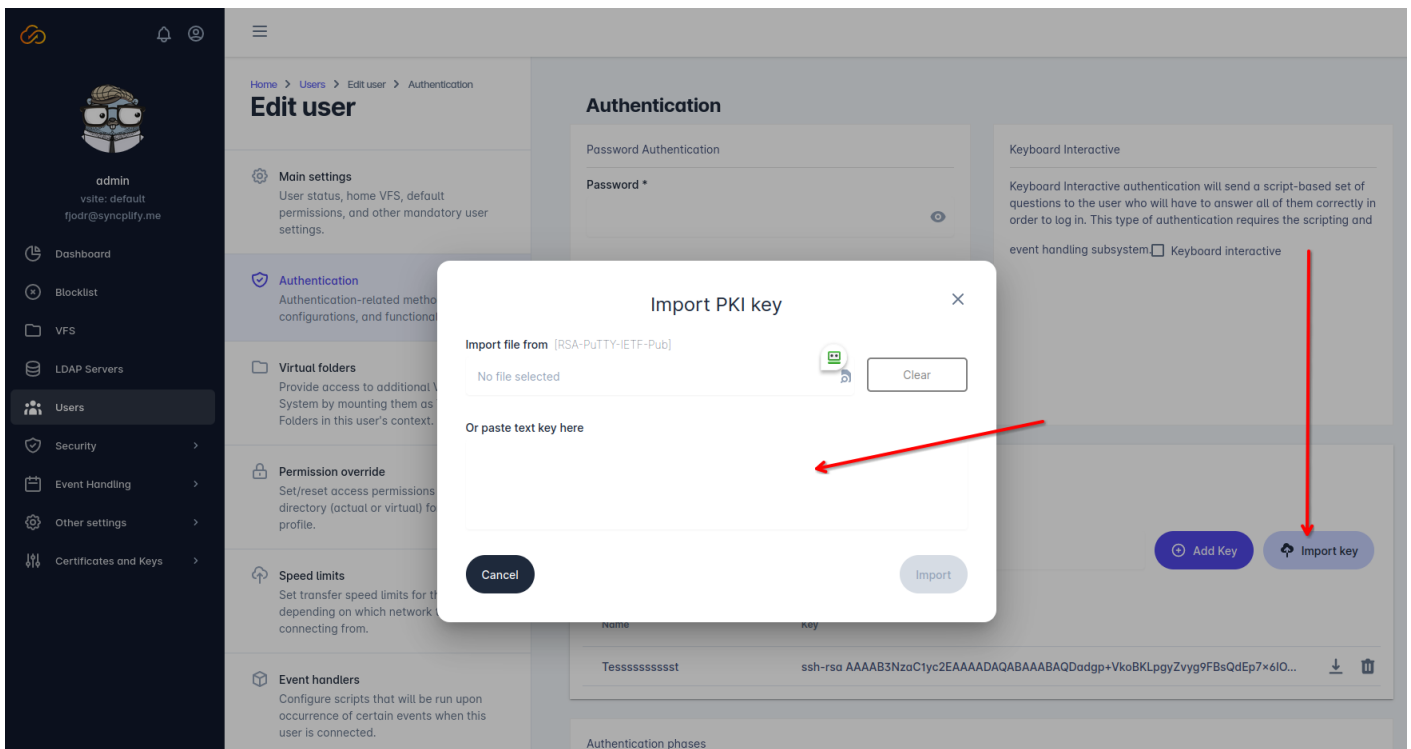
The "Actions" section has three buttons: "Generate a public/private key pair" (with a "Generate" button), "Load an existing private key file" (with a "Load" button), and "Save the generated key" (with "Save public key" and "Save private key" buttons).

The "Parameters" section has "Type of key to generate:" with three radio buttons: "SSH-1 (RSA)", "SSH-2 RSA" (selected), and "SSH-2 DSA". Below it is "Number of bits in a generated key:" with a text box containing "2048".

Regardless of the OS (Linux or Windows) the user needs to **save both public and private keys** after they are generated, and:

- **retain the private key** (will be used in the SSH/SFTP client to connect to the server)
- **send the public key** to the Synclify Server!'s administrator

Once the Synclify Server! administrator receives the user's public key, he/she will have to import it into the authorized keys for that specific user, and enable PKI authentication for that user profile, as shown in the pictures here below:

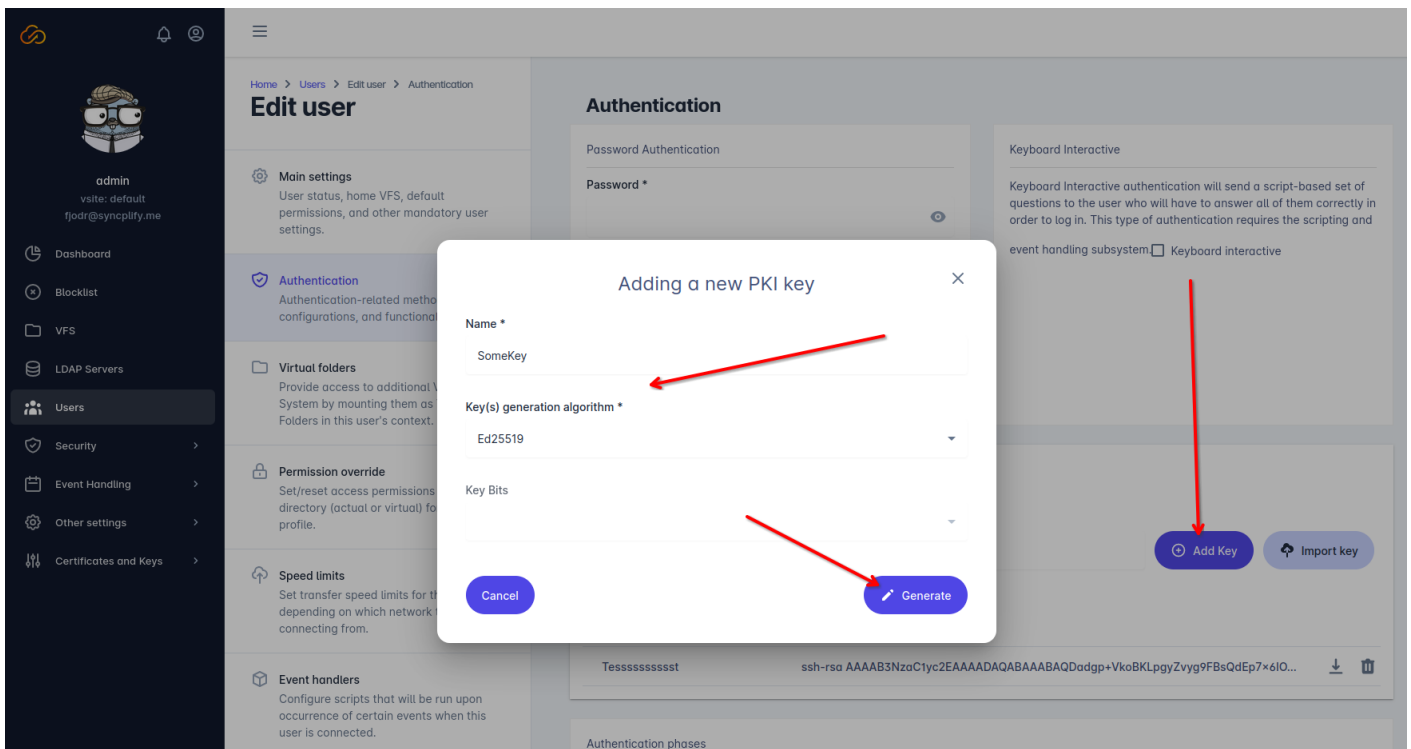


After importing the user's public key, the Syncplify Server!'s administrator will simply save the user profile, and that's it: the user can now authenticate and log in using his own self-generated key pair.

Method #2: server-generated key pair *(not recommended)*

Alternatively the Syncplify Server!'s administrator can generate the key pair and send it to the user, while Syncplify Server! only retains (stores in its configuration database) only the public key. This method is not recommended because it implies the private key being sent over the network, which might be unsafe, but it's much easier than the previous method, and it can actually be suitable and safe in closed environments (like when the key pair is given to an internal user or an employee) in a safe way.

In this case, the Syncplify Server!'s administrator will still need to enable PKI authentication for the specific user profile:



Once done, the Syncplify Server!'s administrator will simply need to save the user profile, and **send/give the generated PPK file to the user.**

Revision #1

Created 19 July 2023 11:26:50 by DevTeam

Updated 22 July 2023 00:51:45 by DevTeam