

# Manual (Batch services)

## Introduction

The purpose of this document is to be a first (technical) introduction to the FIDUS.Brussels (Regional Service Integration) project for new consumers that wish to use batch (SFTP) services.

- [Introduction](#)
  - [About the BRIC](#)
  - [About FIDUS.Brussels](#)
- [ConnectionBatch Services](#)
  - [Connection](#)
  - [Generating a certificate](#)
  - [Connecting to the sFTP](#)
  - [Folder structure](#)

### About the BRIC

The Brussels Regional Informatics Centre (BRIC) is the public interest agency of the Brussels-Capital Region under the supervision of the Minister or the Secretary of State responsible for regional and communal IT.

The BRIC is a development and modernization agency whose remit is to organize, promote and spread the use of information and communication technologies (ICT) among the various Brussels target groups: regional authorities, administrations and agencies; communal institutions; local administrations and authorities; educational institutions; hospitals; citizens.

### About FIDUS.Brussels

Following the publication of the Royal Order of 08.05.2014, the BRIC has been given a mandate to create a regional service integrator for the Brussels region. The goal of the FIDUS.Brussels project is to organize, standardize and facilitate the accessibility of authentic sources. The BRIC offers a platform to exchange information in a uniform and secure fashion through web services. The authentic source owners are regarded as service providers, users are called service consumers. Brussels based agencies can be both service consumer and provider depending on the application and context. Non-Brussels based agencies (e.g. Fedict FSB - the federal service integrator; or BCSS - Crossroads Bank for Social Security) can also offer their own services ,or request information from Brussels based agencies, through FIDUS.Brussels.

As a service integrator, the BRIC plays an import role in connecting government agencies with reduced overhead. Without a central integration service, government agencies would be required to negotiate and implement separate point-to-point connections for each new implementation. With the FIDUS.Brussels platform, government agencies are only required to organize and implement one standardized communication flow to contact all authentic or other data sources offered by other agencies.

The FIDUS.Brussels platform is based on open standards such as SFTP, SOAP, WSDL, XSD, WS-Security, TLS, ..

## ConnectionBatch Services

### Connection

All FIDUS.Brussels batch services are exposed using SFTP. Secure File Transfer Protocol (SFTP) is a secure version of File Transfer Protocol (FTP), which facilitates data access and data transfer over a Secure Shell (SSH) data stream. It is part of the SSH Protocol. All user authentication is done using keys (files contain a private and public key) instead of username, password combinations. Using key based logins with ssh is generally considered more secure than using plain password logins.

Key-based authentication uses two keys, one "public" key that anyone is allowed to see, and another "private" key that only the owner is allowed to see. To securely communicate using key-based authentication, one needs to create a key pair, securely store the private key on the computer one wants to log in from, and copy the public key on the computer one wants to log in to. FIDUS customers need to create their own key pair and send the public key to their FIDUS contact. The public certificates will be used to uniquely identify and trust each consumer.

FIDUS currently has two environments which can be accessed by consumers; staging and production:

- Staging : <sftp://ftp-fidus.irisnetlab.be> on port 22
- Production: <sftp://ftp-fidus.sec.brussels> on port 22

Separate certificates need to be used for each environment.

### Generating a certificate

The easiest way to create the required SSH key is to use OpenSSL. While this tool is available for Windows, the procedure is much more straight forward on a Linux machine. Our example will use a command which will work on any standard Linux box with OpenSSL installed.

The default policy is to create keypairs using 521 bits ECDSA. We will be using a tool called *ssh-keygen* to generate two files: a private and a public key. It is important that only the public key is shared with FIDUS (or others). The private key will be used as your unique identifier, and should be stored securely.

We will need a separate key for each environment. For a new consumer (e.g. *NewConsumer*) who wants to connect to both staging and production, this means running the create key command twice.

To create new keyfiles for the staging environment, use the following command (change *newconsumer* by your own organisation):

```
ssh-keygen -t ecdsa -b 521 -f newconsumer-sta -C newconsumer-sta
```

After running the command, *ssh-keygen* will ask for a password. This is purely for your own security when handling the private key within your organisation. This password should be kept safe and never shared with FIDUS or any other third party.

Once finished, you should have two new files:

- *newconsumer-sta*: private key, must be kept securely
- *newconsumer-sta.pub*: public key, will need to be sent to your FIDUS contact to configure your access to the appropriate environment

To create new keyfiles for the production environment, use the following command:

```
ssh-keygen -t ecdsa -b 521 -f newconsumer-prod -C newconsumer-prod
```

To summarize; after running both commands we should now have two sets of private and public keys. The public keys must both be sent to your FIDUS contact. After they have been added to the FIDUS configuration, you should be able to connect the following endpoints using your preferred SFTP client application:

- Staging : [sftp://ftp-fidus.irisnetlab.be](ftp://ftp-fidus.irisnetlab.be) on port 22 (195.244.165.51)
- Production: [sftp://ftp-fidus.sec.brussels](ftp://ftp-fidus.sec.brussels) on port 22 (195.244.165.52)

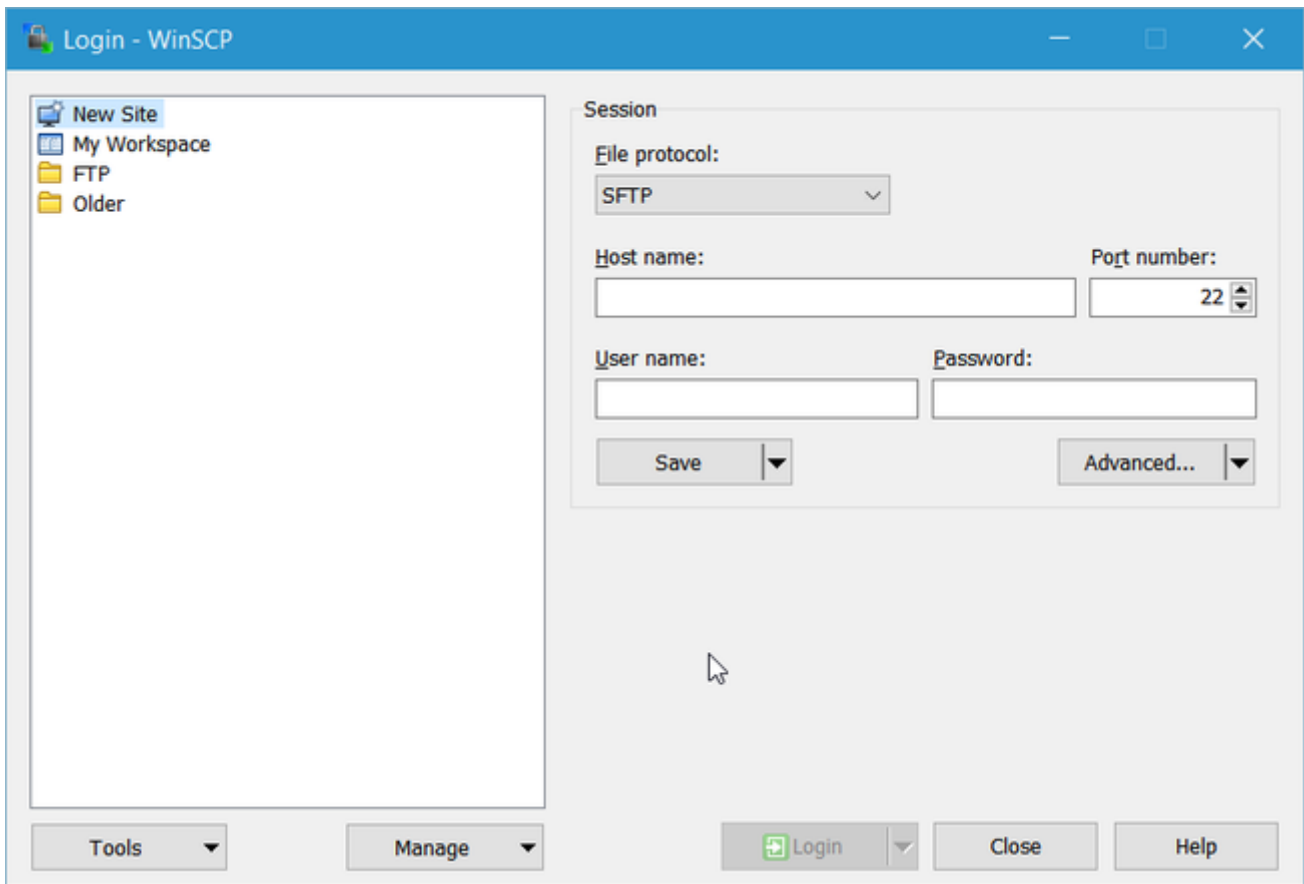
In the following example we will show how one can connect using a SFTP client.

## Connecting to the sFTP

As browsing using a graphical interface is still slightly more intuitive, we will provide a short tutorial on how to test you connection using WinSCP. WinSCP is a popular SFTP client, but other software will undoubtedly work just as well. WinSCP can be freely downloaded from their website: <http://winscp.net/>

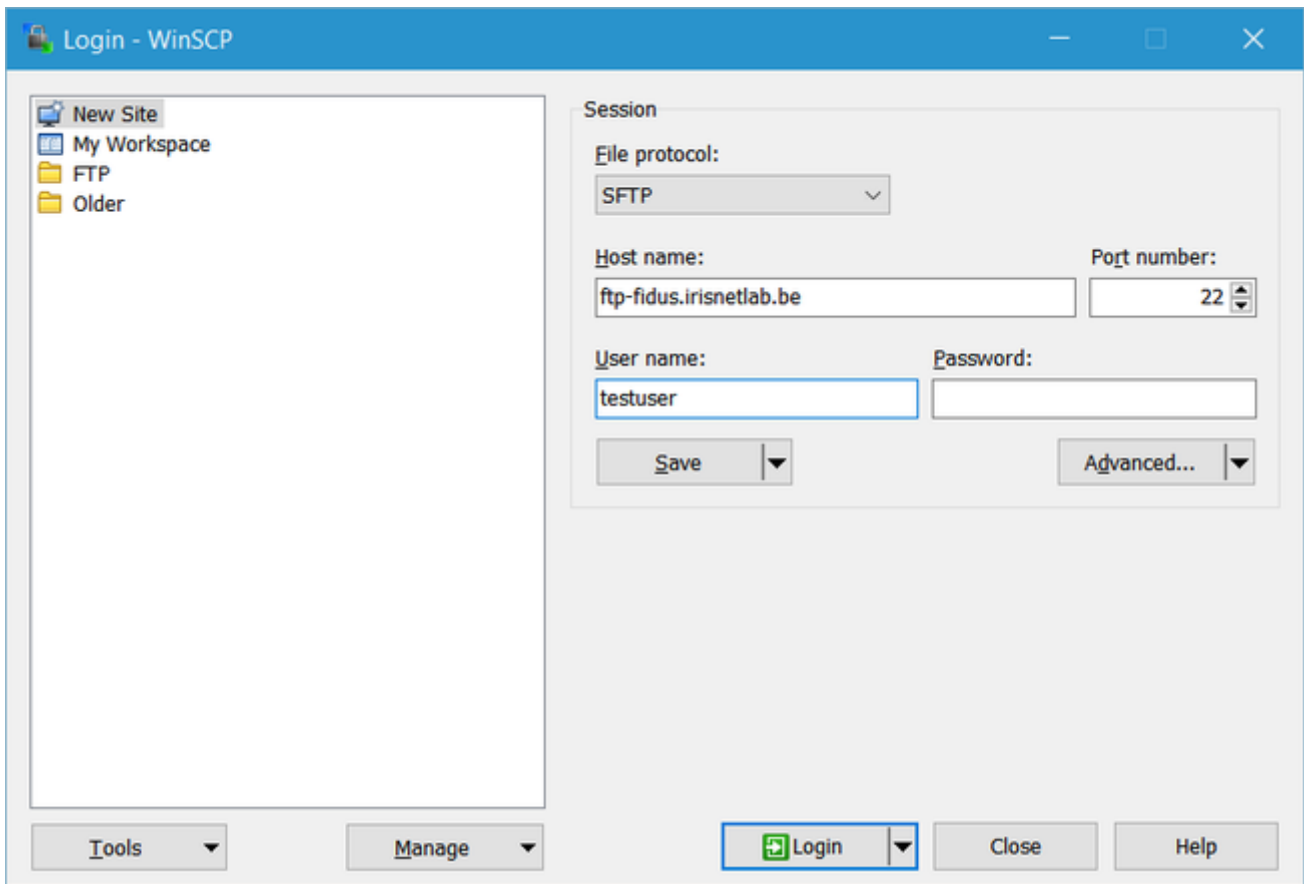
For this example we will connect to the FIDUS staging environment using a keypair specifically created for this exercise: *testuser-sta*.

When we first open WinSCP we see an option to create a new site in the upper left corner. Once selected, we can enter the first details:

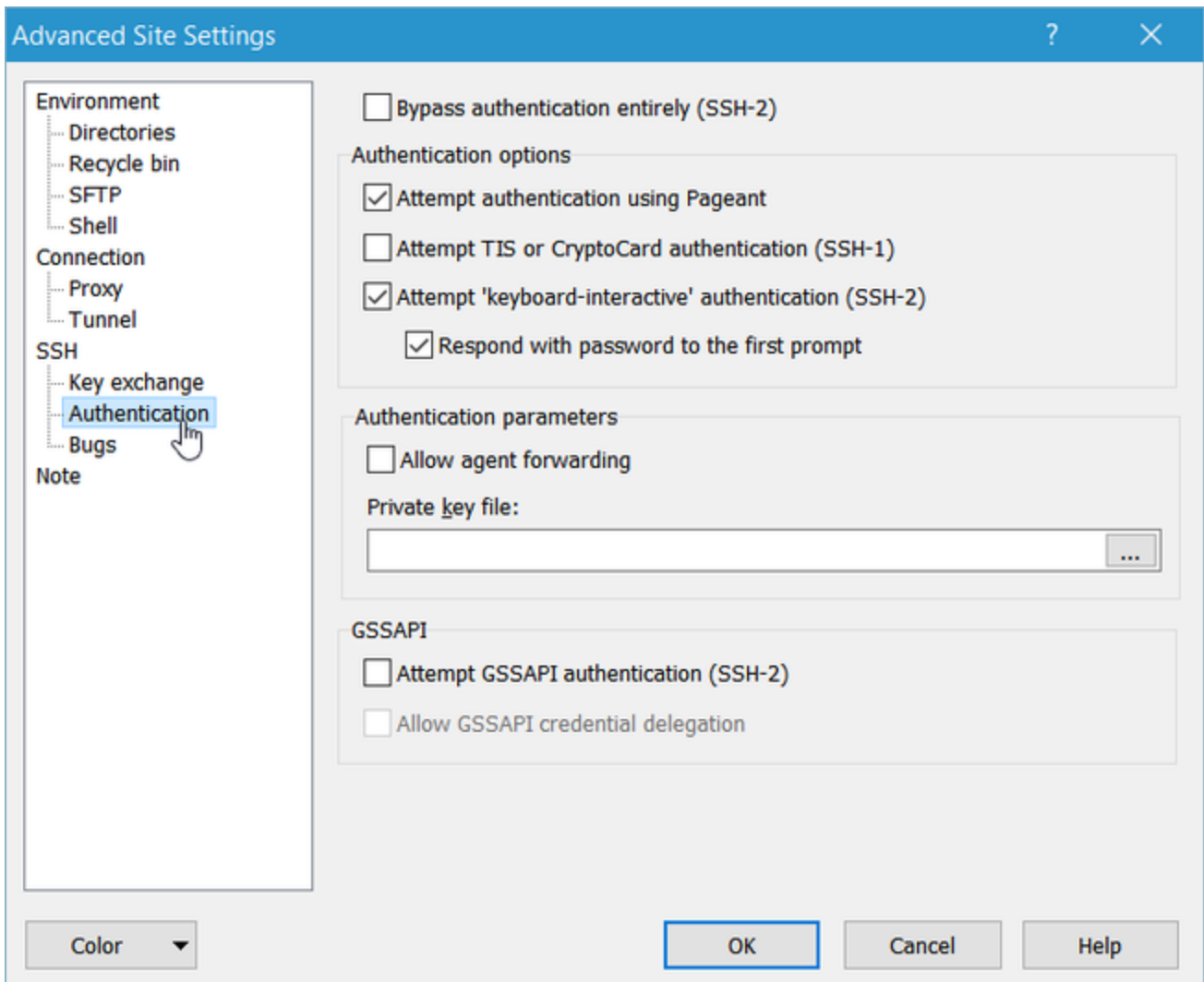


In our case this will be the following:

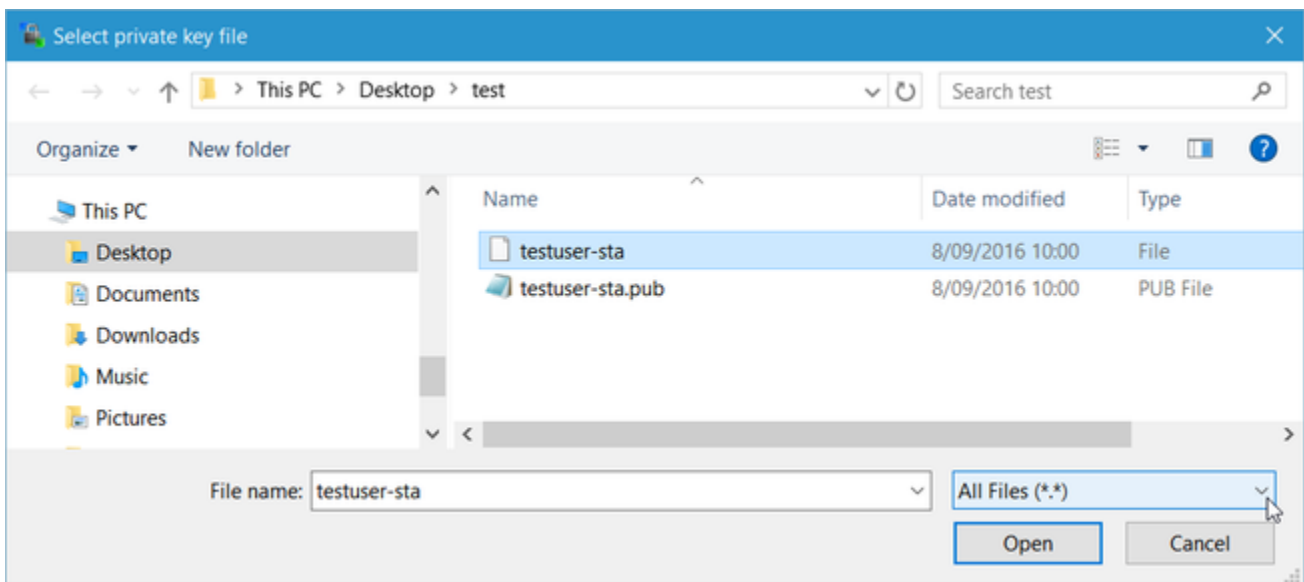
- Host name: <ftp://ftp-fidus.irisnetlab.be>
- Username: testuser



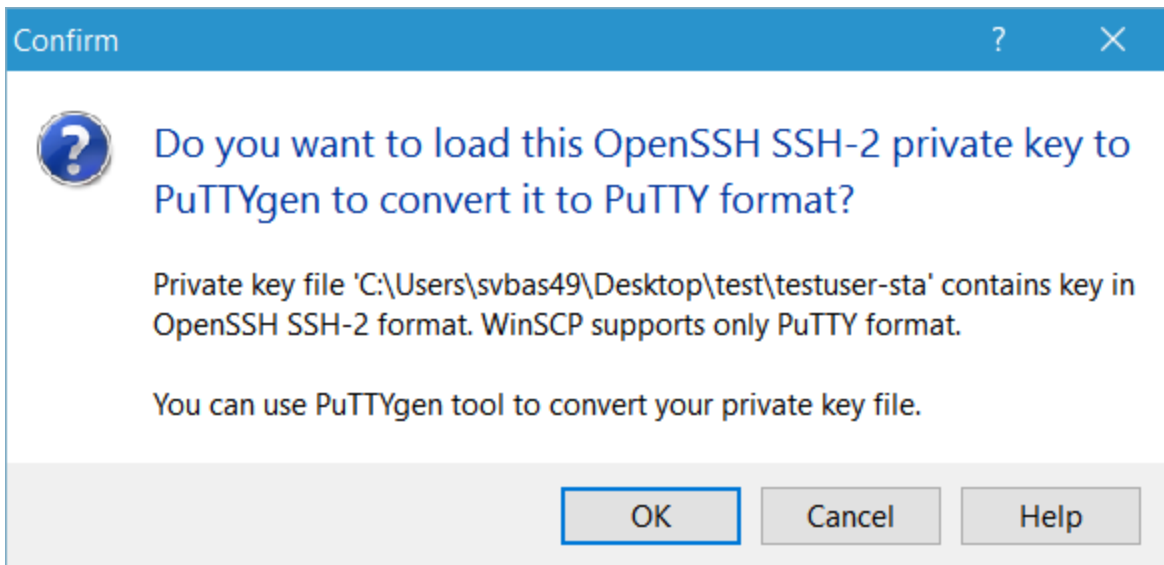
To add the private key which we will use instead of a password, select advanced. In the advanced window, select the authentication tab.



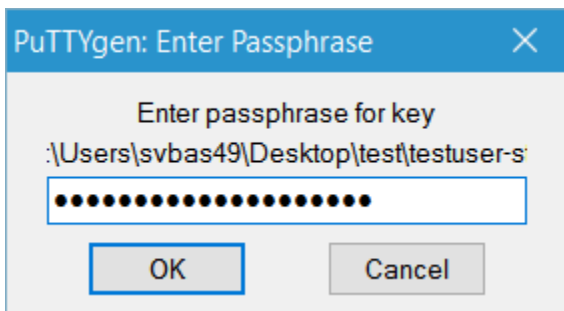
Now select the tree dots next the window labeled 'Private key file'. In the new window, navigate to your staging public key, and set the window to show 'All files'. Select the private key file e.g. *testuser-sta*.



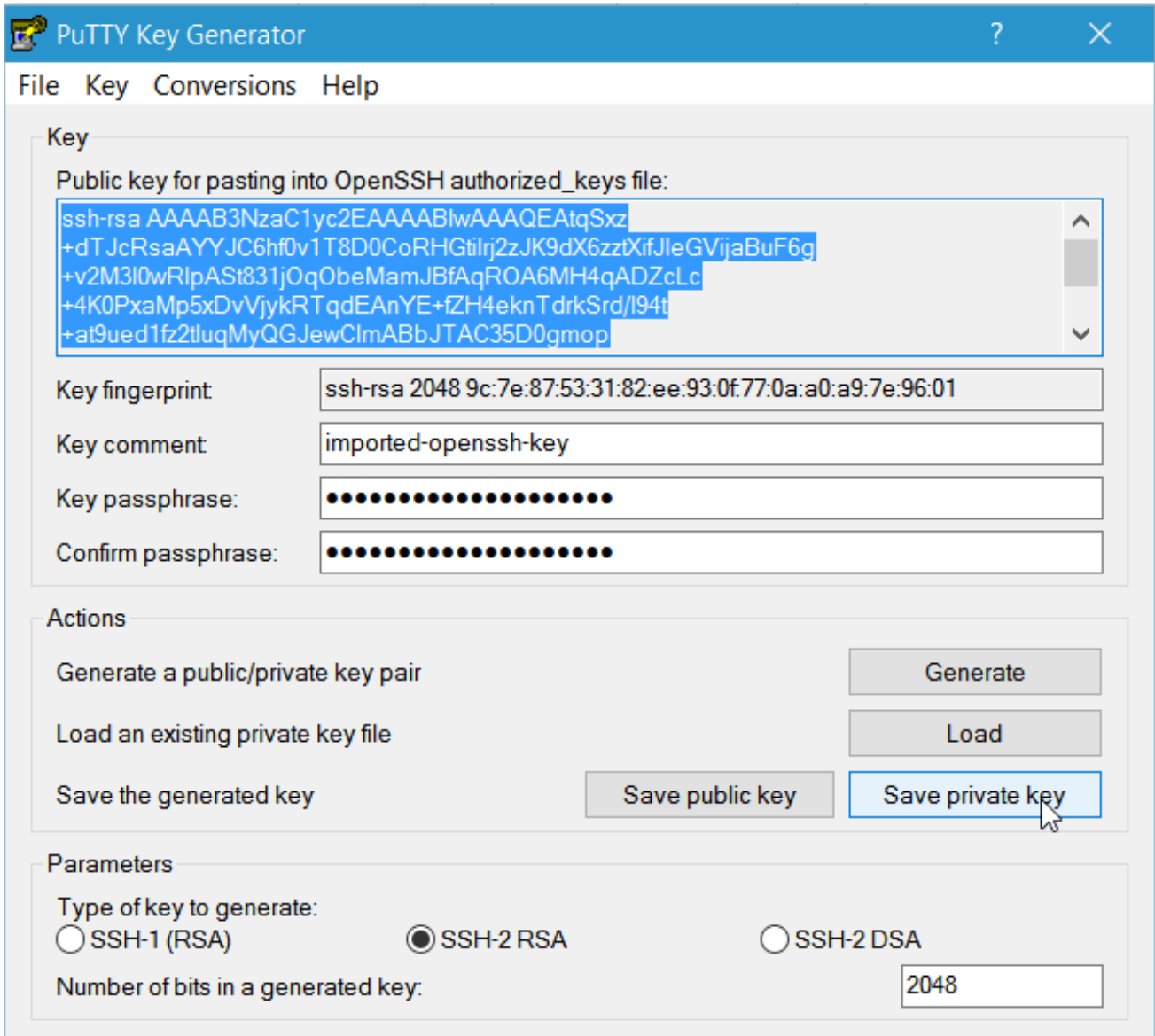
Select open. A prompt will appear to convert your file to the Putty keyfile format, select yes.



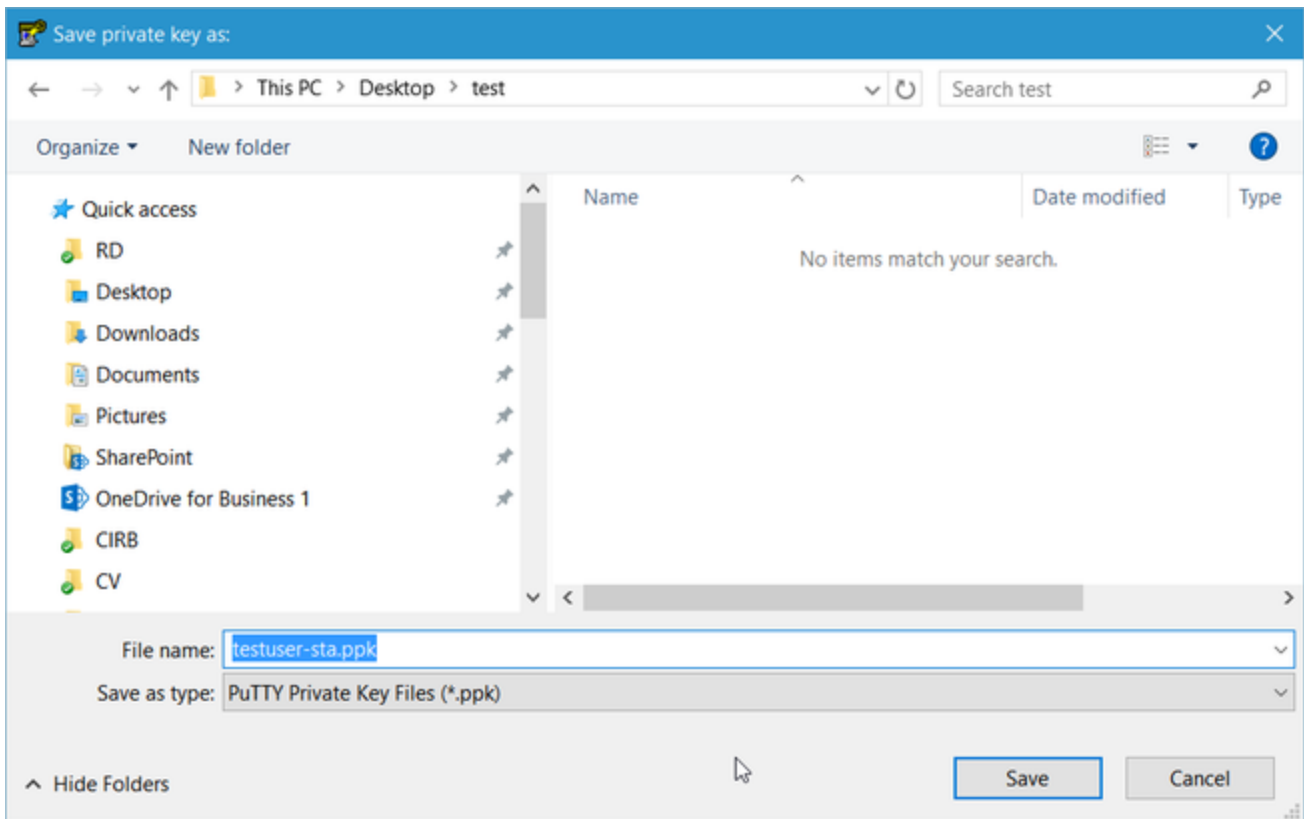
Enter your password if you set one previously.



In the next window select OK to get to the screen where we can save the .ppk file.



Select 'Save private key' and store the file using a .ppk extension e.g. *testuser-sta.ppk*.

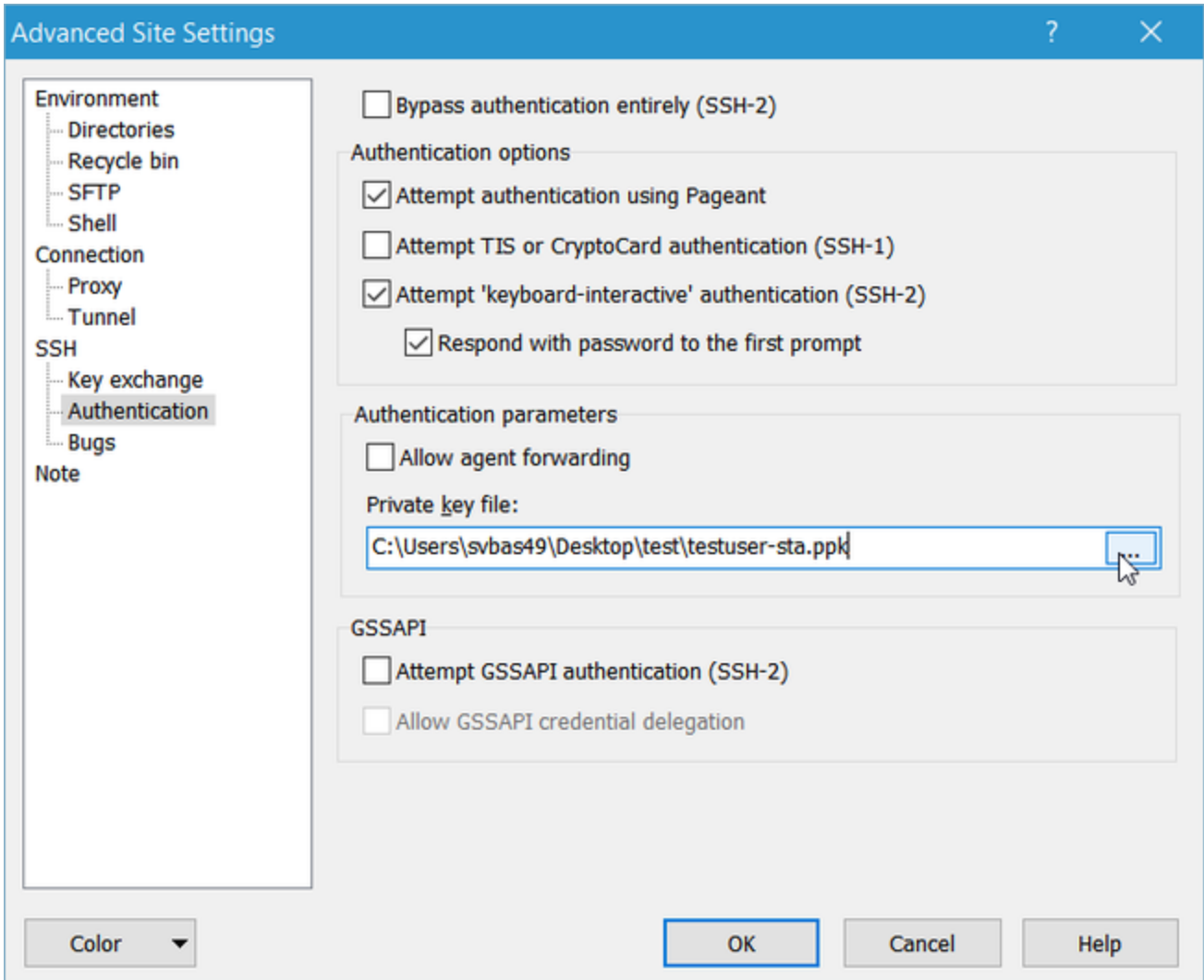


Our folder should now contain three files:

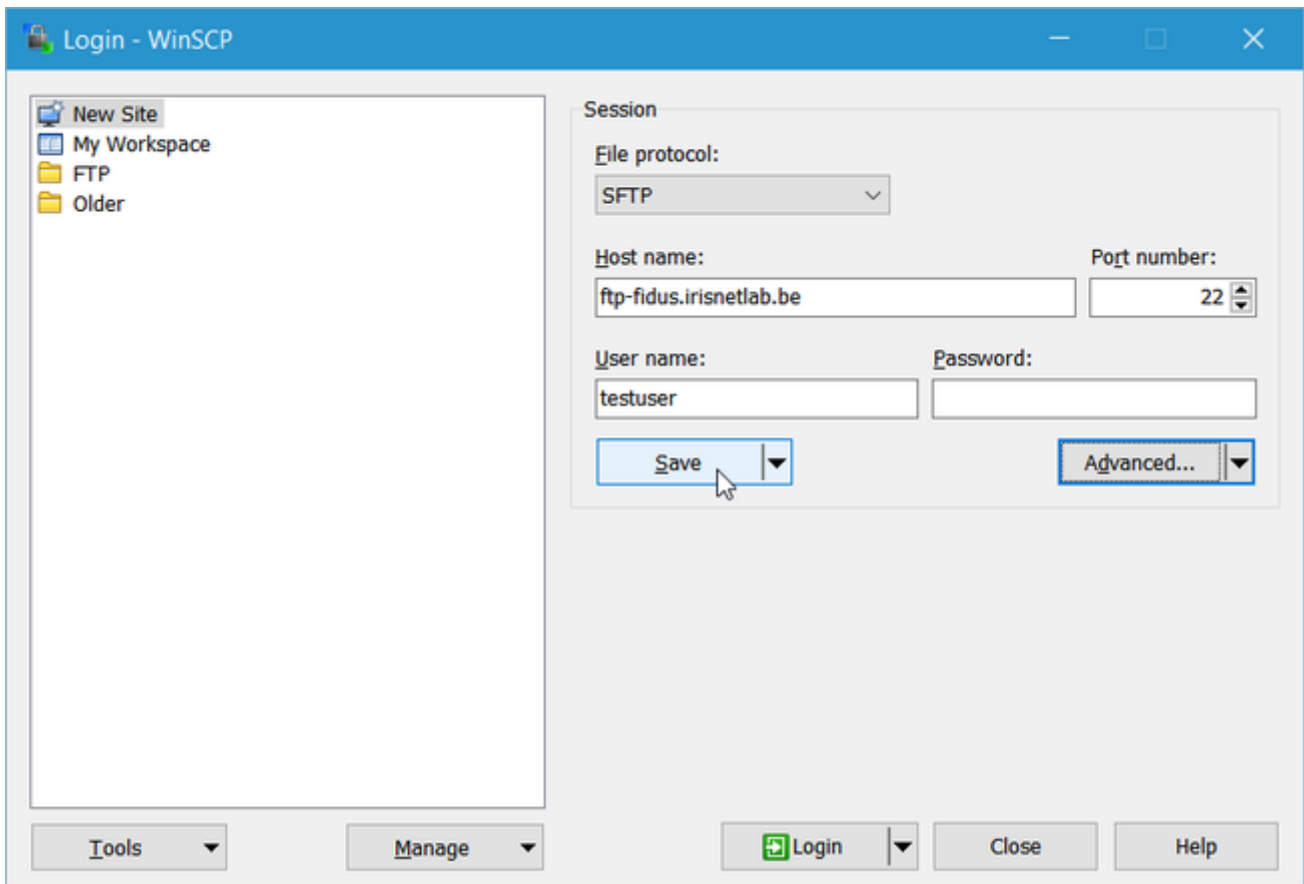
Name	Date modified	Type	Size
testuser-sta	8/09/2016 10:00	File	2 KB
testuser-sta.ppk	20/09/2016 16:16	PPK File	2 KB
testuser-sta.pub	8/09/2016 10:00	PUB File	1 KB

We can now go to WinSCP again to add the .ppk file as our private key:

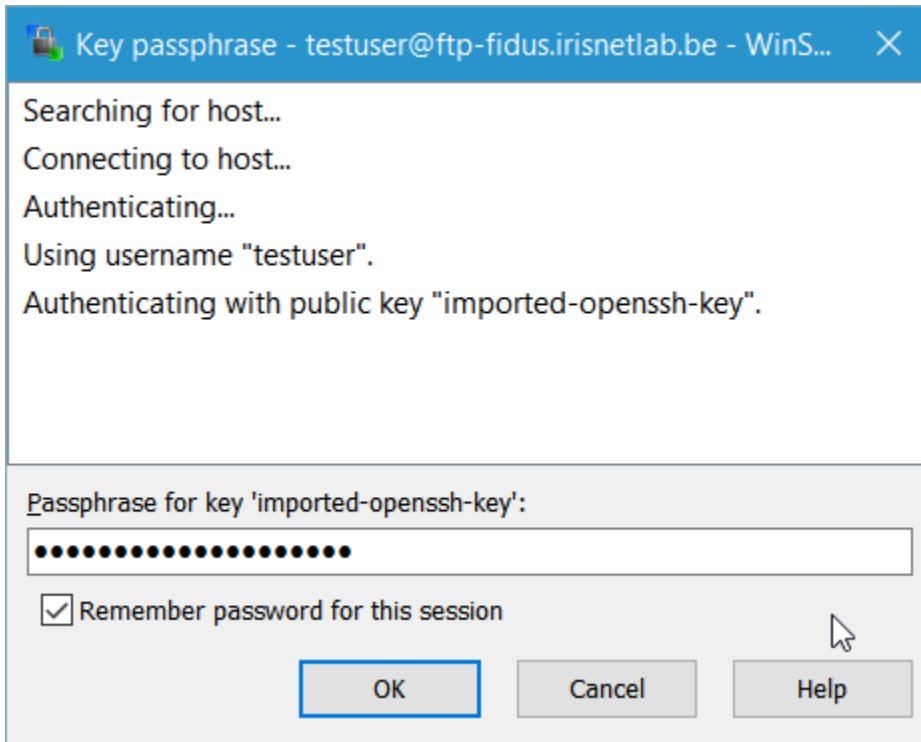




Select OK to go back to the main window. Once there select 'Save' followed by 'Login':



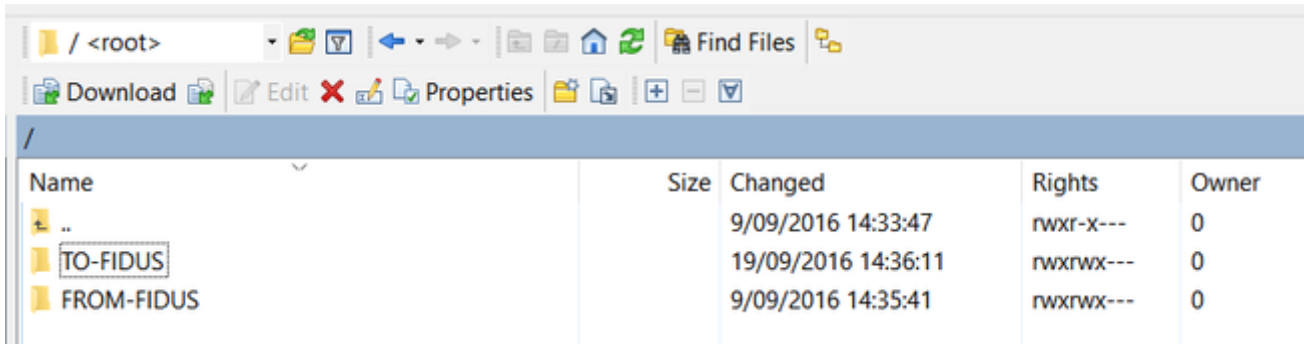
If needed, enter the password for you private key:



If all went well, you should now be connected to the FIDUS SFTP staging environment. For more details, see the following section.

Folder structure

FIDUS batch uses a fairly straightforward method of message exchange. Once a user connects via SFTP, they are automatically taken to their personal home folder. In this folder there are two sub folders: FIDUS-IN, FIDUS-OUT. Using WinSCP the home folder looks as follows:



The screenshot shows the WinSCP interface with the root directory selected. The file list is as follows:

Name	Size	Changed	Rights	Owner
..		9/09/2016 14:33:47	rwxr-x---	0
TO-FIDUS		19/09/2016 14:36:11	rwxrwx---	0
FROM-FIDUS		9/09/2016 14:35:41	rwxrwx---	0

The user has read and write rights in both the TO-FIDUS and FROM-FIDUS folders.

- Files that are meant for FIDUS to process should be placed in TO-FIDUS. Once processing starts they will be removed from the folder automatically.
- Files prepared by FIDUS for the consumer are placed in FROM-FIDUS. The user is responsible for deleting files in FROM-FIDUS once they have been downloaded and securely processed. Caution: once deleted, it may not be possible recreate certain files! Message flows